Programming languages are essential for software development.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
Transpiling on the other hand, takes the source-code from a high-level programming language and converts it into bytecode.  
One approach popular for requirements analysis is Use Case analysis.  
However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages.  
One approach popular for requirements analysis is Use Case analysis.  
Relatedly, software engineering combines engineering techniques and principles with software development.  
Use of a static code analysis tool can help detect some possible problems.  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.  
 Computer programmers are those who write computer software.  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
 Tasks accompanying and related to programming include testing, debugging, source code maintenance, implementation of build systems, and management of derived artifacts, such as the machine code of computer programs.